Visual Analytics for Topic Model Optimization based on User-Steerable Speculative Execution

Mennatallah El-Assady[1,2], Fabian Sperrle[1], Oliver Deussen[1], Daniel Keim[1], and Christopher Collins[2]

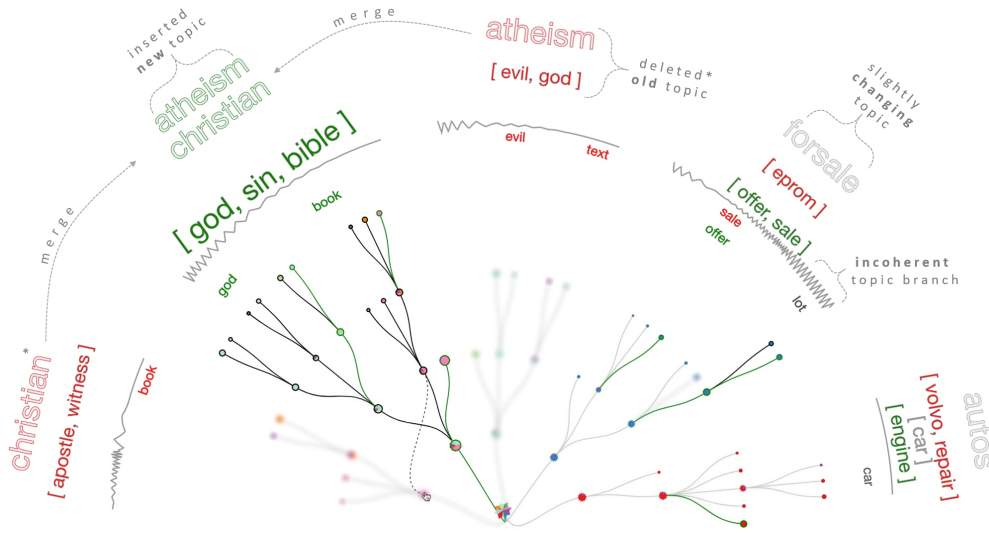


Fig. 1: The *Tree-Speculation View* is used to compare two topic models and shows the differences. Deleted branches are blurred, while **moved**, **newly added** and **removed** nodes and keywords are highlighted. To efficiently guide users towards perceivable model quality improvements, our system automatically proposes optimizations like the merge of two topics depicted here. By visualizing model uncertainties and low quality topics, we foster trust in the model and empower users to directly address these shortcomings.

Abstract— To effectively assess the potential consequences of human interventions in model-driven analytics systems, we establish the concept of *speculative execution* as a visual analytics paradigm for creating user-steerable preview mechanisms. This paper presents an explainable, mixed-initiative topic modeling framework that integrates speculative execution into the algorithmic decision-making process. Our approach visualizes the model-space of our novel incremental hierarchical topic modeling algorithm, unveiling its inner-workings. We support the active incorporation of the user's domain knowledge in every step through explicit model manipulation interactions. In addition, users can initialize the model with expected topic seeds, the backbone priors. For a more targeted optimization, the modeling process automatically triggers a speculative execution of various optimization strategies, and requests feedback whenever the measured model quality deteriorates. Users compare the proposed optimizations to the current model state and preview their effect on the next model iterations, before applying one of them. This supervised human-in-the-loop process targets maximum improvement for minimum feedback and has proven to be effective in three independent studies that confirm topic model quality improvements.

Index Terms—User-Steerable Topic Modeling, Speculative Execution, Mixed-Initiative Visual Analytics, Explainable Machine Learning

1 Introduction

In the context of visual text analytics, topic modeling algorithms are widely used as a processing step to efficiently segment document collections into thematically-related groups [12]. The process of topic modeling remains mostly concealed from end-users to disguise its algorithmic complexity [48, 55]. However, treating topic models as typical black-box machine learning components limits the trustworthiness of their results and, in turn, of the whole system they inform [16]. Tackling the issue of such a one-way process, several approaches have emerged, promoting trust through linking the inputs, outputs, and parameter-spaces of topic models into visual analytics frameworks [67]. These techniques present a step towards truly interactive machine learning through enabling a human-in-the-loop process that allows users to adapt the models to their data and tasks [24]. However, topic models typically operate in a high-dimensional vector space defined by the keywords of a corpus, with documents represented as frequency-distributions in that space. The most commonly used class of algorithms are generative, probabilistic models that create topic-document assignments in multiple optimization iterations. Typical algorithms like latent dirichlet allocation (LDA, [10]) attribute keywords to topics, not documents, which is not intuitive for non-machine-learning-experts to understand, let alone optimize [62]. The main obstacle for designing effective visualizations for such models is the discontinuity between their *parameter-space* and their *model-space*. It is notoriously difficult to get an intuition of the effect of changing algorithmic parameters or directly manipulating the model-space on the output of these models. In addition, "seemingly small changes on the user side could have unpredictable and nonsensical cascading side effects" [47].

In their study on how non-experts perceive, interpret, and fix topic models, Lee et al. recommend that "more active, mixed-initiative support may be useful during the refinement process, such as pointing

[1] *University of Konstanz, Germany*
[2] *University of Ontario Institute of Technology, Canada*
Funded by DFG-777/17, DFG-431/16, and NSERC RGPIN-2015-03916.

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

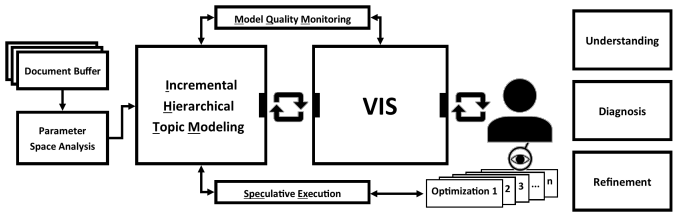


Fig. 2: The user-steerable topic modeling optimization process, including a constant model quality monitoring that triggers speculative optimization strategies, whenever the measured model quality deteriorates.

users to topics with high refinement potential and providing immediate feedback" [47]. Hence, in order to provide complete transparency within a visual analytics system, unveiling the inner-workings and decision-making processes of topic modeling, there is a need for **explainable** and **visualizable** topic modeling approaches [24]. Addressing these challenges, we introduce the **I**ncremental **H**ierarchical **T**opic **M**odel (IHTM), a deterministic, similarity-based algorithm developed to tightly integrate the human in every step of the algorithmic decision-making process. This user-steerable model was developed to maintain a competitive performance with state-of-the-art topic models, while being explainable and visualizable. Furthermore, IHTM is able to request human intervention when required, to efficiently leverage human decision-making. Thus, we present a mixed-initiative visual analytics technique for direct, *bidirectional* interactions between the visualization and the machine learning components.

For effective human-in-the-loop decision-making, we further introduce a novel visual analytics guidance mechanism. Based on the continuous monitoring of the model stress level, the system can prompt users when it detects a problem with the algorithmic decisions, requesting their feedback on a diverse set of automatically prepared model optimizations. To support the user's decision-making, speculations on the effect of possible decisions are presented, allowing them to perform more informed refinements. This **spec**ulative **ex**ecution (SpecEx) highlights the different outputs that may emerge from a user's feedback *before changes take effect*, altering the operational-path of the model. Hence, to estimate the potential impact of an interaction, users can rely on SpecEx as an efficient preview instrument. In this paper, we present a two-fold speculative execution; one on the effect of different optimization strategies, and another temporal speculation on the next steps of the IHTM algorithm.

Fig. 2 depicts our proposed technique for mixed-initiative topic modeling using visual analytics. The document buffer holds batch-segments of the corpus, ready for analysis. These could be coming from a real document stream or a sorted document collection. Based on the parameter space analysis proposed in our previous work [24], every batch of documents is prepared for further processing. The IHTM then sequentially assigns each document to a topic branch, constructing a hierarchical topic structure. Every insertion is directly visualized, evaluated, and can be interactively adjusted. The **m**odel **q**uality **m**onitoring (MQM) component constantly tracks twelve quality measures, evaluating the current *stress-level* of the algorithm. This component informs the IHTM and visual analytics workspace of the current model quality and triggers the algorithm to halt whenever its heuristics detect a decrease in the model quality. The main design rationale behind this process is to target *minimum feedback* for *maximum improvement*.

We have identified several requirements for a visual analytics solution incorporating SpecEx. First, it is essential that the model is understandable, deterministic, and explainable so that users will be able to understand how to interact with it. SpecEx analytics tools require incremental processing, so that users can intervene and steer the model development as the data is processed. In order to enable speculative execution there needs to be some metrics related to the task which can be predicted. In the topic modeling scenario, these are measures of quality. Based on our previous work with experts in topic modeling [24] and motivated by the tasks outlined by Liu et. al [48], we address the following tasks: [T1] topic model **understanding**, [T2] model quality monitoring (**diagnosis**), and [T3] topic-tree **refinement**.

We evaluated our technique with three different approaches, involving 13 participants and over 40 hours of studies. First, we validated the IHTM topic model against common probabilistic models, with all three annotators confirming that IHTM is competitive and achieves results of comparable quality. Second, we verified that experts could use our tool to achieve model improvements through a user study with six participants optimizing a model of a familiar dataset, followed by a semi-structured interview. Finally, we evaluated the model quality improvements by having four annotators rank quality across a variety of configurations, revealing that supervised speculative execution resulted in better models than fully automatic approaches.

Our work makes the general contribution of SpecEx as a semi-supervised optimization method which can reveal possible alternative outcomes for automated processes, to inform model steering. Specifically to topic modeling, our work contributes: (1) the explainable and visualizable IHTM algorithm, including techniques for incorporating the users' domain knowledge throughout the model-building process; (2) a tailored visual analytics workspace for the progressive understanding, diagnosis, and refinement of the modeling process.

2 Background and Related Work

Our work is related to research in four areas: topic modeling algorithms, visualizations, quality measures, and speculative execution.

Topic Modeling Algorithms Topic modeling is a special case of document clustering that adds label generation [1]. Most commonly, topic modeling is understood as a soft clustering problem, i.e., with probabilistic membership assignment of documents to topics. However, other classes of algorithms are also applicable to achieve a meaningful partitioning of corpora. More specifically, four different classes of clustering algorithms can be distinguished: **(1)** *Generative clustering* algorithms are probabilistic and include the popular LDA topic model [10] and its diverse extensions (hierarchical [8], temporal [9], with metadata [60], for spoken text [59]) as well as many other models (Pachinko allocation [52], Author Topic Model [61], HDP [65], pLSI [19]). **(2)** *Spectral clustering approaches* like NMF [40, 68], divide and merge clustering [15] or fuzzy co-clustering [41] perform a segmentation-based clustering based on a dimensionality reduction of the data. **(3)** *Discriminative distance-based clustering* works on concept-, document- or keyword-distances [25, 28]. **(4)** Finally, *word- and phrase-based clustering* incorporates text semantics or linguistic knowledge into the clustering process [5, 32]. Independent of the clustering algorithm used, labels for any topics can be created by specialized algorithms [44, 45]. Probabilistic models often produce results of higher quality at the expense of reproducibility and the understandability of the algorithm. They often require the specification of the expected number of topics before the start of the modeling process. This makes these models harder to use in visual analytics systems promoting understanding and interactivity. As a result, we have devised a novel topic modeling algorithm tailored to the incorporation into a visual analytics pipeline, that will be presented in the following section.

Visual Analytics for Topic Modeling To date, many visualizations of topic model results have been proposed: ParallelTopics [20], LDAExplore [27], VISTopic [69], Hiérarchie [63], and TopicViz [23], among others. Another set of systems does not only interactively visualize the modeling results, but also allows users to change or regenerate the underlying model. Such interactions can add new model constraints to group keywords in a single topic [34], produce more detailed subtopics [39], manually merge or split topics [21, 33], or modify weights for individual keywords [35]. In our previous work, we have optimized two competing topic models by letting users rate their respective results [24]. However, all of these systems recompute at least parts of the model after every interaction. For an effective visual analytics process we enable active steering of the modeling process while it is making decisions, as proposed by Mühlbacher et al. [55], allowing our system to quickly provide feedback on user interactions and avoid waiting times, or even restarting the modeling process. Liu et al. note that "most users often treat a machine learning model as a black box" [48] and have thus defined *understanding*, *diagnosis* and *refinement* as three important tasks for visual analytics systems. Our

workspace addresses these three tasks with tailored visualizations and interactions for different stages of the modeling process. Lee et al. [47] specify refinement operations for topic modeling that are intuitive for non-expert users, including splitting and merging topics or adding and removing keywords that have all been implemented in our system.

Measuring Topic Quality In order to speculate alternative paths to guide the analysis, our system needs a way to differentiate potential modeling strategies. One way to do this is to preview how each strategy will affect the model quality downstream. Assessing the quality of a topic model can be achieved using one of several available metrics or a combination thereof. While each of the metrics studied by Chang et al. [14] addresses some aspect of quality, they found the measures did not capture the human intuition of quality well, thus these measures alone cannot be used to optimize a model toward human expectations. We have implemented twelve different model quality metrics. Some of them measure structural properties of the topic tree, while others operate on the keyword level within the topic descriptor keyword list. **Pointwise Mutual Information** [11] measures the co-occurrence of the most frequent keywords in a topic. Mimno et al. claimed to achieve better performance with their definition of topic **coherence** [53] which measures the conditional probability of keywords given more frequent ones from the same topic. Coherence performs especially well for very small and specific topics as well as large, generic ones [57]. As a complement, we have implemented a measure we call **separation**, which measures the inter-topic separability of keywords of a topic. We aim for coherent yet separable topics and call the ratio between the two metrics **distinctiveness**. In our previous work we have introduced **certainty** [24] as a metric that asserts the uniqueness of keywords for their respective topics. Additionally, we use the **variance** of the similarities between subtopics of one parent-topic as an indicator for topics that only contain equally similar subtopics.

To measure structural properties of the tree we employ graph measures. The 'fan-out' of tree nodes is measured by the **branching factor**. **Compactness**, defined as the ratio between leaf nodes and inner nodes is helpful to avoid document chaining artifacts that introduce sparse and deep trees. In addition, we use the **number of topics**, the average **topic size** [50] and, if available, the number of authors or speakers. Additional metrics include the document entropy [50], the word mover's distance [42], and the topic distribution size. As these are focused on probabilistic models, they are not implemented in our system.

Speculative Execution Speculative execution, in parallel computing (e.g., [13,58]), is used to reduce latency in I/O tasks by pre-fetching and executing probable instructions. In visual analytics, we propose to adapt speculative execution to provide real-time previews of multiple paths to guide user decisions. Gleicher et al. argue against too much guidance and seek to foster serendipitous interactions for exploratory tasks [2]. For the model optimization task, however, a more guided approach may be helpful. Lee et al. introduce preview mechanisms for topic modeling [47]. Building on this, our speculative execution provides previews of alternative paths for topic model optimizations.

3 Incremental Hierarchical Topic Modeling

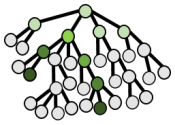
We introduce the **I**ncremental **H**ierarchical **T**opic **M**odel (IHTM) (Algo 1) as the foundation of our mixed-initiative visual analytics technique. This explainable model progressively builds a hierarchical topic-tree by iteratively inserting each document from a collection. The leaf nodes of the tree represent individual documents, while all inner nodes group these documents into topics. The root node is thus an aggregation of the entire corpus. The document-inserts are based on a monotonic similarity-function, making the output of the algorithm deterministic, with each document belonging to exactly one topic. Documents contributing to multiple topics can be segmented into sub-documents using a topic-based document split routine [26], with each sub-document being inserted into one topic. Cosine similarity is our default similarity function, but can be exchanged to suit the task and genre. Documents are represented as weighted keyword-vectors, as proposed in our previous work [24]. The keyword-vector $V \in \mathbb{R}^w$ of each inner-node i is the sum of the vectors of all its leaf-node children $l(i)$: $V_i = \sum_{c \in l(i)} V_c$. Here, w is the number of keywords in the corpus.

Algorithm `IHTopicModeling`(*document-buffer*)

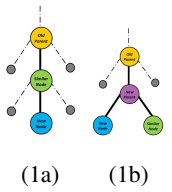
```
root = initialize the topic-tree with an empty root;
while document-buffer is not empty do
    node = create node from next doc; // incoming node
    if numKeywords(node) ≥ minKeywords then
        similarNodeSet = recursiveTreeDescend(node,root);
        updateTree(node,similarNodeSet);      // insert
        extractTopicDescriptors(root); // update topics
        updateMQM();      // calculate all metrics
    else unclusteredNodes.add(node) // doc too short
    updateVisualisation();   // show insert decisions
    if qualityDeclined() then
        optimize();          // trigger optimization
    incorporateUserFeedback();       // bidirectional
applyFinalTreePruningAndSegment(); // best tree cut
return extractTopicDescriptors(root);
```

Algo 1: Incremental Hierarchical Topic Modeling

When inserting new documents, the algorithm determines the similarity between the new document node and existing tree-nodes based on the similarity of their respective keyword-vectors. Starting with the root, the model finds the most similar branch within the tree for inserting the incoming node. To guarantee finding these branches without having to compare every node to the whole tree, we rely on the monotonicity property of the similarity function [4]. It is defined as: $\forall d,p,c \subseteq T : min(sim(d,p),sim(d,c)) \geq sim(d,p \cup c)$, with d as *incoming document-vector* and p,c as direct *parent* and *child* nodes in the tree (T), respectively — $p \cup c$ is the normalized sum of the vectors of p and c. This guarantees that the similarity cannot increase when going from leaf nodes to the root, making each parent node in the tree more general than its child. Hence, for every new insertion, the algorithm starts with the root, calling a recursive insertion function `recursiveTreeDescend`. It evaluates the similarity of the incoming node to the current node and all its direct children. If one of the children is found to be more similar than the current node, the algorithm descends its respective tree-branch, otherwise recursion is terminated.



The result of this similarity computation is a set containing a single most similar node, or a set of equally-similar nodes, called `similarNodeSet`. The model then distinguishes between three cases to complete the insert in the `updateTree` routine, as described in the following. **(1)** If the incoming node is similar to one other node, it differentiates whether the similar node is an inner-node or a leaf-node. In case of an inner-node, the incoming node gets appended as a direct child (1a). Otherwise, a new mutual parent for the incoming node and the similar leaf-node is created to ensure that every document remains a leaf-node in the tree (1b). **(2)** If multiple sibling nodes are equally similar, the incoming node is appended depending on whether these are the only children of their parent or only a subset of the children. For the former, the node gets inserted as another sibling node (2a), and as for the latter, the node and all similar siblings get inserted under a new parent (2b) (which becomes a direct child of the previous parent of these siblings). **(3)** If the tree has more than one equally-similar node and not all of them are siblings, heuristics determine the most likely insert branch, selecting one node (e.g., closest time-stamp or same author) and proceeding with the first insert strategy.



We measure the quality of the model after an insert by calculating multiple quality metrics introduced in Sect. 2 (`updateMQM()`) and evaluate the last insert position using the *insert certainty*:

$$\text{insert\_certainty}(d) = \frac{\max_{n \in \texttt{similarNodeSet}} (\text{sim}(d,n))}{\left(\frac{\sum_{n \in m} \text{sim}(d,n)}{||m||} \right)}$$

with m as the set of all nodes visited by `recursiveTreeDescend`. It measures how certain the insertion into a topic was by comparing

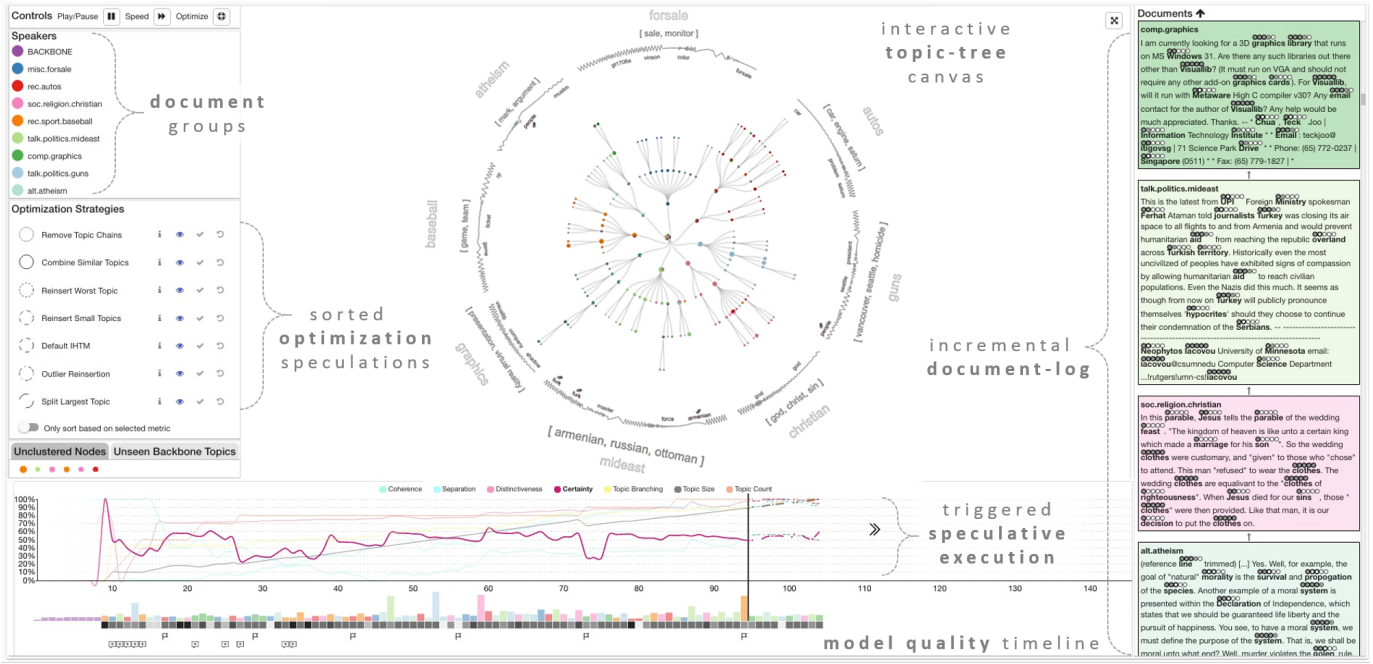


Fig. 3: The visual analytics workspace. New documents are added to the *document-log* on the right-hand side before they are shown in the *topic-tree*, the central component of our workspace. It shows both the model space and the algorithms decision making process, as well as topic hierarchies, keywords and uncertainties. The *model quality timeline* tracks the metric development over time and forecasts the effects of speculative executions (see Sect. 5). A *control panel* on the left hand side ranks optimizations based on their effectiveness for the current model and enables steering the speculative execution: Optimizations can be viewed 👁, accepted ✔ and rejected ↺. The *control panel* on top allows pausing the modeling process ⏸, setting its speed ⏩, and manually triggering a speculative execution 🎯. A color map shows which *document group* the nodes in the tree belong to. *Unclustered nodes* that have not yet been added to the model are listed and can be added via drag and drop.

the similarity to the chosen parent node with all other similarities encountered during the insertion process. This triggers an update of the visual analytics workspace (`updateVisualization()`), as discussed in the next section. As our system is uncertainty-aware, it automatically pauses the computation whenever necessary, requesting the users' feedback. The IHTM treats all direct children of the root as high-level topics and uses a scoring function (log-likelihood ratio [49]) after each document-insertion to extract the top keywords for these topics as *descriptors* (`extractTopicDescriptors()`).

Following the recommendation of Lee et al. [47], our model is designed to be an explainable topic modeling approach and support all their suggested update and refinement operations, including adding, removing, and changing keywords; moving documents within the topic hierarchy; and lastly, splitting, merging, removing, and creating topics. Thus, the IHTM can be tightly coupled with a visual analytics system that constantly interacts with the algorithm throughout its progressive model-building process. IHTM is based on an iterative, distance-based clustering technique that yields a visualizable *model-space* — in contrast to typical probabilistic, generative topic models. As documents are incrementally inserted into the tree, the IHTM does not require the specification of an expected number of topics before the start of the modeling process. This is an important property for the use of topic models in a streaming context or when visualizing every model-decision step iteratively, as is described in the next section.

4 Visual Analytics Workspace

The visual analytics workspace is the primary window of interaction with which the users have to steer the topic modeling process. Supporting this complex task, we strive for a single screen interface to reduce the users' cognitive-load during the analysis. As motivated in the introduction, the visual analytics workspace supports three analysis tasks, namely **[T1]** Topic Model **Understanding**, **[T2]** Model Quality **Monitoring**, and **[T3]** Topic-Tree **Refinement**. Thus we designed different visualization components, tailored to aid in performing these tasks. Fig. 3 shows an overview of the visualization interface, including all views. In addition to this interface, panels on each side of the screen open up on demand, enabling users to adjust parameters (settings-bar, right) and see details of interactively selected items (details-bar, left).

Depending on the current state of the topic modeling, the workspace enters one of two modes, varying the functionality of the views and, in turn, the possible interactions. First, the **operation-mode**, is the primary state in which the IHTM is running, updating *one model*. Second, the **speculation-mode** is a state where the IHTM is waiting for the user's feedback, showing *candidate-models* as alternatives that the user can compare and choose from. While the users can incorporate their feedback in both modes, the type of expected interactions and feedback is different. In the operation-mode, only one instance of the topic-tree is available for interaction. Therefore, the workspace supports more fine-grained, direct tree manipulations, as described in Sect. 4.2. In contrast, in the speculation-mode many different candidates trees are available as possible optimizations of the current tree. The central target for the interaction in this mode is the comparative analysis of two trees. This is supported by tailored difference-visualizations, e.g., Fig. 1. This section focuses on describing the visual workspace while in the default operation-mode, followed by Sect. 5 that describes the system state while in speculation-mode.

Design Rationale To achieve a simple yet expressive design, we incorporated a set of visual cues throughout the interface. The first and most important cue is the *highlighting of keywords and descriptors* in all visualization components. This provides the users with an intuition of the *unit of analysis* considered by the topic modeling, i.e., users should become aware that topic models reduce documents to vectors of keyword frequencies. To highlight this, we designed *keyword-glyphs* that encode the global (topic-model relevance score; cf., [24]) and local (frequency in the document) importance score of keywords. Both scores are discretized onto a five-point scale, encoding the global score to the outside of the glyph (black circle-border) and the local score to the inside of the glyph (gray circle), e.g., ●●●●○ (global score 1, local score 4) and ○○○○○ (global score 5, local score 2) . These glyphs are positioned on top of keywords in several view components,

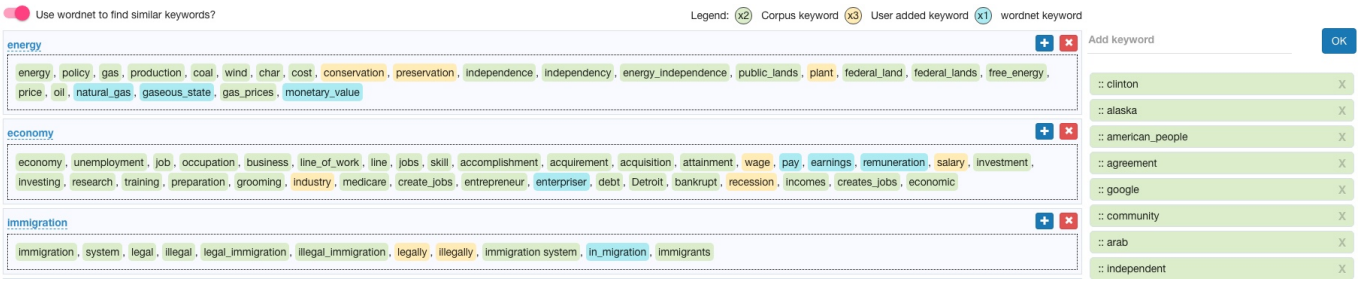


Fig. 4: To prime the model with their domain knowledge, users can provide a topic backbone consisting of expected topics, each characterized by representative keywords. On the side we list the most important keywords of the corpus, from where they can be dropped into the topics. Keywords are colored based on whether they originate from the **corpus** or have been added manually or through **word2vec** as similar keywords.

as in Fig. 3. They are used, for example, on hover, in detail-bars, and in the *document-log* — our close-reading [37] view.

In addition, as the IHTM computes uncertainty information for nodes, edges, and insert-operations, we designed *uncertainty-aware visualization* components using a concept of *visual-fuzziness*, as introduced by Vehlow et al. [66]. The different visual elements encoding this uncertainty information will be introduced within the remainder of this section. Lastly, we strive for *complete transparency* when displaying the results of the modeling. We, therefore, include information on unclustered documents or backbone topics that users expect but that have not been detected by the algorithm in a panel for investigation.

Interaction Design All visualization components are highly interactive to support the users' exploration and analysis. For example, as shown in Fig. 3, each document is represented in three coordinated views in the workspace: in the document-log, in the topic-tree (as a leaf node), and in the timeline. In addition to *linking-and-brushing*, we provide *two levels of details-on-demand* for topics and documents. First, a detail-panel on hover that contains only essential information such as keyword distributions, second, a detail side-bar with all relevant quality measures and other statistics. To further reduce complexity, the workspace is initialized with sensible *default parameters* for all components. However, we also provide an *expert-interaction mode* that shows a wide range of controls for more advanced adaptations of the visualization, or of the algorithm. Lastly, to ensure data-provenance [64] tracking and *trust-building*, we track and show all user interactions, displaying all instances of optimization and direct model-manipulations in the timeline view (see Sect. 4.3).

4.1 IHTM-Backbone Priors

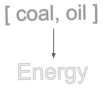
As motivated by Andrzejewski et al. [3], introducing "must-link" and "cannot-link" constraints to topic models yields substantial improvement as it introduces a user-defined notion of relevance for the semantic of the document-collection at hand. However, Lee et al. found that this method of specifying complex semantics is unintuitive to users [47]. Jagarlamudi et al. [36] suggest allowing users to generate topic keyword lists to direct models. The IHTM algorithm was, therefore, designed with this concept of domain-knowledge incorporation in mind. Based on our experience working with social-science scholars [24] on topic model tuning, we deduced that one potential place to include users' feedback on their expected topics in a corpus is *before starting* the model-building. Hence, we introduce topic backbone priors as a technique to prime the model, avoiding a cold-start. Fig. 4 shows the interface in which users can construct the topic backbones. They can create a hierarchical topic structure through adding topics. These could be derived from an external data source (e.g., a discussion-agenda), or could be automatically computed by our system using heuristics and other topic models. For example, users can provide some representative documents and determine a topic distribution using LDA [10]. They can then directly adapt the suggestion in-place, through editing the keywords manually, declaring topics as irrelevant, or completely removing them. They are additionally assisted by an automatically-ranked keyword list that extracts the most discriminative keywords based on the G^2-metric [22]. Similar keywords could be computed using *Wordnet* [51] to support the users in generating a comprehensive backbone. Users can specify names for the backbone topics, which are visualized in the topic-tree, as discussed in Sect. 4.2.

The backbone technique allows the users to adapt and guide the IHTM to their preferred topic granularity. For example, when analyzing news articles, users might choose to create a high-level backbone (sports, technology, crime, etc.), or they might prefer a more specific one about concrete events (a particular game, new phone, etc.). This allows the users to adjust the topic modeling process to their data and tasks for a more targeted analysis. The backbone keywords are weighted and used to initialize the IHTM topic-tree before starting the document insertion-loop of the algorithm. However, these backbones are weighted-in merely as subtle indicators for the model. Hence, if the model recognizes documents as not being similar to any user-defined backbone topic, it will construct a new topic branch. Therefore, users cannot accidentally manipulate the model into generating nonexistent topics. Unseen backbone topics are indicated in the visual interface to guide the user (see Fig. 3). Finally, any backbone topics that users have created can be saved and loaded back in the next session, enabling users to create collections of their domain knowledge that can be modularly applied to their data.

4.2 Model-Space View

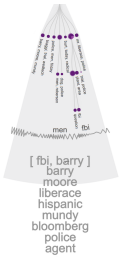
As described in Sect. 3, the IHTM algorithm updates the visual workspace on the internal state of the model and all performed tree-update operations after each insert-step. Visualizing the model-space of the algorithm, the radial topic-tree is incrementally built as a central part of the workspace, as depicted in Fig. 3. To guarantee scalability, the tree is placed on a zoomable canvas and users can collapse branches interactively to hide topics or remove tree imbalances. The tree-root represents an aggregation of the entire corpus, while leafs are documents, and the inner-nodes represent the topic-hierarchy.

The default color scheme of the topic-tree mirrors the distribution to document-authors (or utterance-speakers for conversation transcripts) per node. The node colors can be changed to show the inner-node variance. In addition, the size of each node represents the amount of text accumulated in all of its associated children. Applying the metaphor of node-fuzziness [30], we encode the within-cluster diversity, which we described to users as uncertainty for simplicity of explanation, of each topic (variance of pairwise-similarities of its children) to differentiate nodes with potential for improvement from certain topic-nodes. The same concept is applied to branches using *squiggly-lines* on the outside of the tree. Line-segments showing higher frequency and amplitude indicate *noisiness* or higher variance in sub-topic branches. In addition to such confidence-indicators, the outside of the tree shows three levels of keywords and descriptors, as shown in Fig. 5. The keywords closest to the tree (on the inside of the line) correspond to the second-level sub-topics in the tree, i.e., the different branches of the main topics, which are, in turn, all direct children of the root.

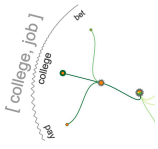


Due to space constraints we only show these keywords for larger topics, and not for every leaf node. The main topic descriptors are represented as an array on the outside of the line, e.g., [coal, oil]. Users can choose to name each topic. These names are then shown on the outermost segment, e.g., *Energy*. Besides getting the topic names directly from the user, the

algorithm derives names for the topics based on defined backbone priors (Sect. 4.1). To reduce visual clutter, the topic tree only depicts the top keywords and descriptors for topics and sub-topics, respectively. Keywords appearing in multiple topics are highlighted with a glyph. However, these first few keywords of the topic vectors might not reveal the nuanced differences between their content. We, therefore, integrate a rich set of interactions. This includes hovering over keyword vectors and topic names to reveal more descriptors. In addition, all nodes in the tree show a detail-box on hover to enable understanding why they were positioned in their respective branch in the tree. Furthermore, by clicking an element on the topic tree, a detail-panel is opened from the left side of the screen.



Our visual analytics framework visualizes the process of topic modeling incrementally. Therefore, we use staged-animations to show each document-insertion into the topic-tree. Each document enters the workspace as a new textbox in the document-log. Using an animated transition, a node is formed out of this box and traverses down the tree, from root to leaf. The visualization shows all decisions the algorithm is taking during the recursive similarity computation as described in Algo 1, line 5.



Starting from the root, similar candidate branches are highlighted in green (darker color showing more similarity) indicating the decision-path of the IHTM algorithm. In addition, the most similar nodes along this path are highlighted. After the insert-position of the incoming node is found, we employ a clock metaphor and place it as the rightmost child of its new parent. The document-circle is animated into its new position and the tree updates all keywords and node-positions. This step-by-step animation directly explains the inner-workings of the model. To speed up the modeling process, users can adjust the speed of the animation or skip the branch highlighting phase, ensuring scalability to larger corpora.

While observing the model building process, users can stop the algorithm to directly interact with the topic-tree and incorporate their feedback. This can be done through dragging-and-dropping any node in the tree (or any unclustered node) to a new parent. To assist this process, the system highlights on demand the outlier children in a sub-topic or similar nodes to a selected one, enabling users to move outlier children to a new branch, or bring similar nodes from other branches together. If users find a document or a sub-topic in the tree that is incorrectly classified, they can trigger an automatic delete-and-reinsert action. This positions the deleted node and all its children at the front of the processing queue, forcing the IHTM algorithm to reinsert them into the current tree. This interaction is particularly useful to overcome earlier errors resulting from the incremental model-building process.

4.3 Model Quality Monitoring

Monitoring the quality of the topic modeling is a vital guidance-element of our visual analytics approach. The IHTM algorithm issues an update for the twelve model quality measures during every insert-loop (Algo 1, line 8). To visualize and interact with these measures, we include a timeline view, as depicted in Fig. 3. This timeline shows every inserted document as a bar (scaled to its text length and colored like its corresponding tree node), and indicates its insert-certainty as small square below. The darker such a certainty-square, the higher the uncertainty of the insert. In addition, to provide provenance histories, we note all interaction events with icons on the bottom of the timeline. These events include the creation of a new topic, manual movement of nodes by the user, a node reinsert requested by the user, and the start of a new speculative execution cycle. Furthermore, all user-selected measures are visualized as line-charts. Due to the changing scale of the measures during the incremental topic modeling process, all scales are automatically stabilized, normalizing the measures between 0% and 100%. To facilitate comparison, all measures are adjusted to be optimized towards 100%, i.e., a rising line for a measure indicates an improvement in quality. One measure can be highlighted (as the tracked measure), fading-out the color for the other lines. Hovering over any

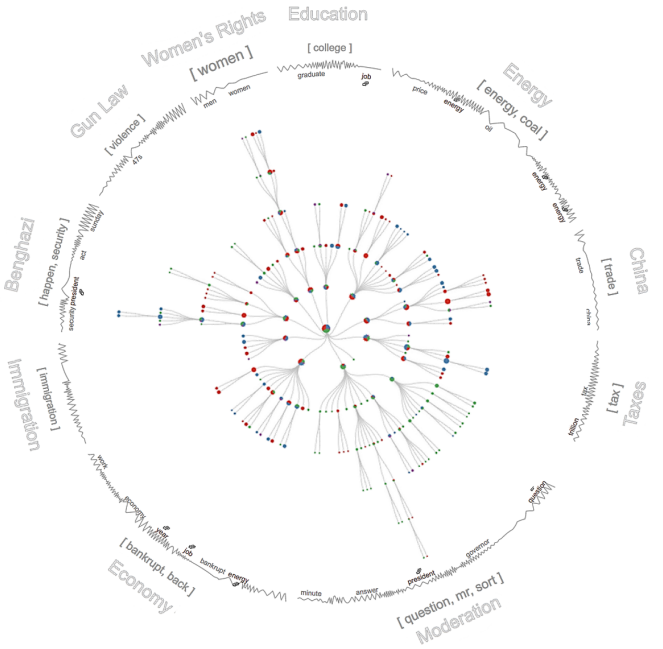
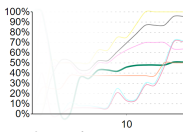


Fig. 5: IHTM result (with backbone; supervised SpecEx optimizations; no direct tree manipulations) for the second presidential debate between Romney and Obama in 2012 [17] in our topic-tree visualization.

position on the timeline shows details of all the measured scores for the respective document, associated with the selected timestamp. Lastly, as quality measurements are typically unstable for the first documents, we fade-in the measures after the eighth document has been inserted to avoid misinterpretations. Through interaction with the MQM users inherently perform a *multi-objective optimization* of all metrics to their *subjective notion of quality*.



5 Speculative Execution

As described in Sect. 2, speculative execution is a method that has long been used in software engineering and process optimization [58]. With our work, we showcase the application of SpecEx in a human-in-the-loop decision-making context using visual analytics. More specifically, for the use-case of topic model optimization, Lee et al. advise that "one possibility to allow users to manage cascading side effects is if the system can provide an estimate of the potential impacts of a refinement before it is applied." [47] Following their suggestion, we present a two-fold speculative execution within the model-building process of IHTM. First, the speculation shows the impact of different optimization strategies, allowing users to compare their effect before deciding on accepting one of them. Second, a temporal-speculation shows the further development of different versions of the topic-tree over several processing steps. Combining both preview mechanisms provides a powerful tool for efficient, mixed-initiative topic model optimization.

Such a speculation can be manually triggered by the user at any point during the incremental topic modeling. However, for a more targeted optimization, we introduce an automatic triggering technique that prompts the topic modeling to halt when its quality deteriorates. The IHTM algorithm continuously assesses the change in its model quality after each insertion. As described in Algo 1, line 11, an optimization can be automatically triggered when a decline is detected. To measure this decline, we have defined and implemented 16 trigger strategies, and evaluated their performance against a set of manually selected trigger points on two datasets. These triggers consider moving averages, as well as various groups of quality metrics. Some heuristics showed promising results, with both high precision and recall values for detecting a decline within a window of +/- 2 documents from the manually-specified points. We, therefore, chose to combine the two best-performing heuristics as default triggers for SpecEx. These are:

"Majority of Visible Metrics Decreases" (MVM) and "Rapidly Falling Slope for Any Visible Metric" (RFa), reacting to quality declines of more than five percent each, for all metrics in the timeline, and an increasing negative delta for any metric, respectively. In combination, they achieve an f-score of 0.706 (precision of 0.667 and recall of 0.750) when tested against the manually annotated corpora.

When triggered, the speculative execution stops the algorithm and starts optimizing parallel copies of the current model state. Hence, at the time of speculation, there are multiple model candidates available, as highlighted in Fig. 2. These candidates are generated by applying different optimization strategies on copies of the latest topic-tree, as introduced in Sect. 5.1. Simultaneously, the visual analytics workspace switches from operation-mode to speculation-mode, activating comparative instances of the different visualization components, as described in Sect. 5.2. The user is then prompted to compare the different optimized models to chose one of them. When a certain optimized model is accepted by the user, the current model instance is replaced with the chosen model. Our system guides users by ranking the optimizations according to their potential in improving the current model quality based on all measures (or optionally on the tracked metric). However, choosing the optimization that yields the highest measured improvement does not necessarily coincide with the users' perceived model quality, so exploration is suggested.

5.1 Automatic Optimization Strategies

As shown in Fig. 3, our framework implements six optimization strategies. These have been developed based on observations of manual optimization intentions. Each of these strategies gets automatically applied to a copy of the current IHTM model. The visual analytics workspace also includes the default IHTM model to allow the users to compare the temporal-speculation of keeping the current model running without optimization. In the following, we explain all options users can compare and choose from during speculation-mode. (0) **Default IHTM**: Shows the future of the current model without adjustments. (1) **Combine Similar Topics**: Combines the most similar topics by moving them to a common parent node. (2) **Reinsert Worst Topics**: Identifies the *worst* topic under the currently tracked metric and redistributes its documents. (3) **Reinsert Small Topics**: Reinserts all documents belonging to topics with three or less documents. (4) **Outlier Reinsertion**: Identifies and reinserts *outlier-nodes* with the algorithm used when creating the *squiggly-lines* in the visualization. (5) **Split Largest Topic**: Splits the largest topic (by number of documents) into two new topics. (6) **Remove Topic Chains**: Identifies *topic-chains*, a clustering artifact, and moves the affected nodes to a common parent.

5.2 Comparative Speculation-Mode

In contrast to the operation-mode (Sect. 4), the main focus of the *speculation-mode* is to facilitate *comparative* analysis. Users can efficiently *flip through*, compare, and interact with multiple optimized models, without any of these candidates affecting the model-building process, unless explicitly confirmed. This section describes the transformation of the topic-tree and quality-timeline from single-model visualizations into comparative views.

Timeline-Speculation View Whenever a speculative execution is triggered, the timeline shows a preview of the effects of choosing one of the optimization strategies. In addition, supporting the two-fold speculation, the timeline also shows how the measures would be affected over the next ten insert operations for each optimization. This preview relies on loading the next ten (buffer-size) documents from the IHTM-buffer, for each optimization, in parallel. For a more distant look into the future, users can request more batches of ten documents from the buffer (by clicking ») or increase the buffer size. As seen in Fig. 3, all speculative measure developments are shown after the trigger-line (which indicates the time-point the algorithm was stopped). These measure-indicators get updated whenever the user selects another optimization strategy to compare. Hence, this

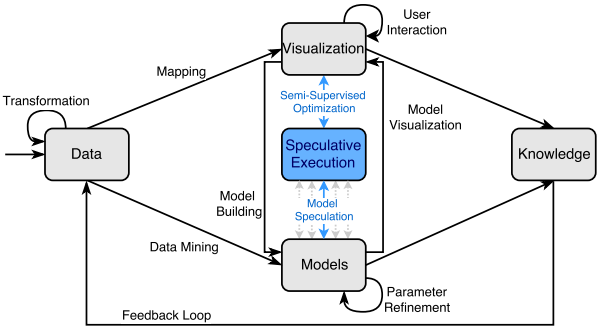


Fig. 6: Extension of VA-Pipeline by Keim et al. [38] to **SpexEx-VA**, with speculative execution as semi-supervised optimization technique.

view allows users to estimate the effect an optimization will have on every measured quality metric, before accepting such an optimization.

Tree-Speculation View Comparing two topic-trees in the same view is a challenging task, as attested by previous approaches, e.g., TreeVersity2 [31] or CandidTree [46]. In contrast to methods proposing juxtaposed views [56], we aimed at designing a comparative tree visualization that comprehensively highlights the differences between two models within the same topic-tree to reduce the cognitive load, as advised by [29]. As shown in Fig. 1, our proposed *tree-speculation view* highlights the *changes* an optimization strategy introduces to the current model-state. Such differences include: split, merged, deleted, or newly-added topic-branches; changes on the document-level; or keyword changes. Deleted branches are blurred, while **moved** (from a deleted branch), **newly added** (including 'future documents' from the temporal speculation) and **removed** branches and keywords are highlighted, using different colors, respectively. This design focuses the attention of the users on the newly created (optimized) topic-tree while promoting the tracking and spotting of changes. To further aid the users' understanding of the ongoing tree-transformation, we design a set of tailored interactions. Notably, hovering a deleted or a moved node highlights its new position in the tree and connects it with its original position. An example of such an interaction is shown in Fig. 1, where the (old) topics of 'Christian' and 'Atheism' were merged into a new topic through moving their high-level topic nodes under a new, joint parent-node. In order to focus the analysis task of this view on comparison, most interactions from the topic-tree component are disabled. Hence, users cannot perform any direct tree-manipulations while in speculation-mode. Once an optimization strategy is accepted, the tree-speculation view transitions into a topic-tree view and all interactions of the operation-mode get enabled again.

5.3 SpecEx as a Visual Analytics Technique

Our instantiation of SpecEx can be seen as an extension of the Visual Analytics Pipeline [38] into a mixed-initiative guidance approach. When there are modeling decisions to make, or multiple models to consider, SpecEx provides an additional bridge between the models and the visualization, allowing for model speculation and leading to a semi-supervised optimization, as shown in Fig. 6. Speculation is possible across a number of different analytic options, such as different algorithms, temporal predictions on streaming data, alternative parameter settings, or multiple runs of non-deterministic algorithms. Speculation may be useful in scenarios where deeper model understanding is required to build trust, such as in explainable machine learning scenarios. Thus, SpecEx can reveal the relevant, alternative pathways considered by a model to enhance user insight into its inner-workings.

6 Evaluation

Due to the modularity of our framework and the subjectivity of defining good quality topic-models, we chose to perform three independent evaluations, each focusing on one relevant aspect of model quality. The first study targets the evaluation of the IHTM algorithm, allowing us to establish a baseline for the other studies. The second study is a qualitative expert study with political scientists, linguists, and computer

	Annotation Score [1-10]			Improvement to LDA		
Algo	News [18]	Debate [17]	Avg	News [18]	Debate [17]	Avg
LDA [10]	5.9 (0.5)	5.6 (0.7)	5.7 (0.6)	0%	0%	0%
HDP [65]	**7.9** (1.3)	7.5 (1.2)	7.7 (1.2)	**35.4%**	32.9%	34.3%
hPAM [54]	6.7 (0.8)	8.0 (0.5)	7.4 (0.6)	15.2%	42.1%	28.6%
IHTM	7.7 (0.7)	**8.4** (0.8)	**8.0** (0.8)	31.1%	**49.4%**	**40.3%**

Table 1: Comparison of IHTM against state-of-the-art topic models. Annotators scored the topics (scale: 1–worst, 10–best; standard deviation in parenthesis). The improvement is relative to the LDA baseline.

scientists. Lastly, the third study is a controlled quantitative study to assess the relative model quality improvement of using our system against the IHTM algorithm baseline.

6.1 IHTM Algorithm Evaluation

To assess the correctness and perceived quality of the basic IHTM algorithm, we conducted a comparative study based on expert-annotations from three political scientists. For this study, we chose two corpora with different characteristics, namely, (1) a collection of 200 news articles from the COHA-corpus [18] (sorted by timestamp), and (2) a transcribed presidential debate [17]. Where applicable, the models where restricted to 50 and 10 topics, respectively, based on previous knowledge of the data. All the experts were familiar with the content of the two datasets. We asked the participants to rate the quality of each topic (per model) on a scale from 1 (worst) to 10 (best). We gave them printed sheets each containing the output of one topic run (on one dataset). The sheets were randomized and anonymized. The annotators reported their judgements were influenced by *whether they could find all expected themes in separate topics*, and based on the *number of meaningful keywords per topic descriptor*. This study compared the result of the IHTM topics (no optimization, no interaction) with three other probabilistic topic models, HDP [65], hPAM [54], and LDA [10]. These were chosen as a baseline, since they have been previously considered as reference models in other comparative evaluation studies [7]. Furthermore, as our evaluation was based on expert feedback on each topic model result as a whole, we did not consider other clustering techniques that do not perform a cluster-labeling step (as these would require a time-consuming comparison of cluster-memberships).

As shown in Table 1, the IHTM output achieved an average score of 8, outperforming the three other models. On average, annotators rated IHTM 40% better than the LDA baseline (an improvement from 5.7 to 8.0), and 3.9% better than HDP (an improvement from 7.7 to 8.0). The average inter-annotator agreement was 84%. IHTM performed best on transcribed discourse data, as this type of text has a linear-structure based on the social interaction between speakers. Such data aligns well with our incremental processing approach. These results indicate that the basic IHTM algorithm produces topic modeling results that are competitive with state-of-the-art-models.

6.2 Qualitative Results: Expert Feedback

This study focused on participants using our framework to optimize a topic model. We recruited six different participants from different disciplines: two experts each from linguistics, political science, and computer science. Each participant completed a two-hour session, in three parts: (1) interface-explanation and first-impression feedback [30 mins], (2) using the visual workspace to optimize a topic model [60 mins], and (3) a semi-structured interview on the design and usability of the system. All participants performed their analysis on the transcript of the second US presidential debate between Romney and Obama in 2012. This dataset was chosen for the familiarity of its content and due to the corpus length (approx. 280 utterances). To set the same baseline for the model-building process, all participants began with predefined backbone priors that included all topics on the agenda of this debate [17]. During all study sessions, the default automatic triggers for speculative execution were used. Finally, all sessions were audio-recorded, screen-captured, and logged for later analysis. In the remainder of this section, we report aggregated feedback gathered from all sessions combined. Fig. 5 shows one example of a topic-tree regenerated based on the settings of the study.

Initial Feedback After explaining the task of the study and the goals of our system, all participants immediately recognized the added value of a visually-explainable topic modeling process. In particular, the trust-building aspect was consistently mentioned in the feedback from most participants. Some of them rely heavily on topic-modeling for their research, building complex statistical models based on the results of topic models. For example, *Pol2* reported that "[he] usually invest[s] an extensive amount of time to manually refine the results of the topic modeling, because [he] has to ensure that the foundation of [his] statistical computation is solid to avoid wrong inferences." He continued, that "[he] usually know[s] exactly which topic distribution to expect for [his] datasets, as [he is] the one collecting the [experimental] data." In particular, this participant was enthusiastic about the idea to set backbone priors for guiding the algorithm. The same general sentiment was shared by other participants who agreed that setting a "semantic-frame" (*Ling2*) to the analysis is intuitive and is expected to yield an improvement in the modeling results.

Observations During the Optimization Process We witnessed different patterns across the groups of participants. For example, the linguists were tempted to analyze the keyword distributions in the topic-tree more closely than other participants. This resulted in them using direct tree manipulations to group different keyword clusters at the beginning of the optimization process, extending the idea of priming the algorithm with semantic knowledge. They also reported relying on the document-log and the topology of the topic-tree to judge the quality of the model. In contrast, the two computer scientists were much more focused on choosing optimizations that did not negatively affect the measured model quality. One of them admitted at the beginning of the session that he would trust the system recommendations more than his own intuition. However, through interacting with the system and understanding the model decision-making process, this user became more skeptical and started comparing different optimizations based on his understanding of the data. He later reported, that "[he] would have more confidence in the results of such a transparent process that [he] can understand, than in a hidden model." Lastly, a general pattern observed in the analysis of the political scientists is that they both were very keen on understanding the specifics of the quality measurement to select "some important ones" to focus on. During the analysis session with one of them, we observed that he, too, relied on the automatic triggers to stop and judge the model quality. However, in contrast to the computer scientists, his choice of an optimization strategy relied mostly on a concrete action-plan which he derived from analyzing and comparing the topic-trees. While these usage patterns are not indicative or generalizable to research fields or broad user groups, observing the differences showed the broad applicability of our system. Regardless of the analysis strategies, they all reported perceiving an increase in the model-quality, even when some of the quality metrics decreased. Furthermore, they praised the rich set of interactions provided in the framework and the amount of details provided.

General Assessment After going through a topic model optimization process, all users confirmed that they perceived an improvement in the model quality. When asked about whether they performed the three tasks targeted by our framework, all of them confirmed that they did. In addition, some of them said that they also did exploratory analysis, and gained insight about both the dataset and the modeling process. They additionally stated that they had built trust in the system: not only did they learn to rely on its guidance (to request feedback when the model quality declines), but most of them also learned to have more confidence in judging when the algorithm is making wrong decisions (e.g., because of missing semantics). They all approved of the SpecEx process, indicating that it is intuitive to use and that it helped them in making informed decisions. Furthermore, some experts also suggested additional requested features to be added to the framework. Notably, a score measuring the impact of the topic-backbone on the model result, a score indicating the differences between two topic-trees in speculation-mode, and an option to jump back in time and change previous decisions, allowing to undo operations. All of these suggestions are potentially useful extensions to our framework.

Corpus	SpecExec	User Rank	Avg. Change	Coherence	Separation	Distinct.	PMI	Certainty	Branching	Compactness	Topic Size	Speakers	Topics
Presidential Debate [17]	no	3.0 (0.8)	16.2%	+5.95%	**+0.86%**	+7.24%	**+8.44%**	**+5.19%**	+29.82%	**+19.76%**	+6.16%	+63.64%	0%
	automatic	2.75 (0.96)	22.74%	+10.34%	-12.73%	-2.10%	-4.20%	+0.85%	+54.09%	+1.79%	+51.37%	+80.30%	-29.41%
	supervised	**1.00** (0.00)	**32.50%**	**+26.40%**	-16.02%	**+14.10%**	+1.89%	-4.65%	**+68.10%**	+18.20%	**+65.13%**	+87.33%	-35.29%
AP Corpus [6]	no	3.00 (1.41)	141.38%	-33.07%	+21.79%	**-2.63%**	-31.45%	-38.97%	**+492.52%**	**+206.52%**	+333.33%	+368.00%	-76.92%
	automatic	4.50 (1.91)	**172.16%**	-54.65%	**+39.18%**	-10.00%	-45.10%	-41.33%	+457.14%	+150.09%	**+550.00%**	+582.50%	-84.62%
	supervised	**1.00** (0.00)	104.41%	**-24.58%**	+1.45%	-18.56%	**-28.23%**	**-37.65%**	+297.96%	+148.77%	+286.10%	+317.30%	-74.36%
20news Corpus [43]	no	2.25 (0.50)	43.50%	+11.37%	+2.44%	+15.59%	-0.53%	**-7.87%**	+135.93%	+14.59%	+157.74%	+67.14%	-61.54%
	automatic	4.25 (1.71)	**80.35%**	**+22.91%**	**+4.21%**	**+35.18%**	**+6.68%**	-9.04%	**+231.02%**	**+89.57%**	**+236.30%**	+114.73%	-69.23%
	supervised	**1.00** (0.00)	66.24%	+13.80%	-0.29%	+15.68%	+2.05%	-9.85%	+206.94%	+59.04%	+210.87%	+97.32%	-69.23%

Table 2: Relative quality metric change when applying a topic backbone and using SpecEx, compared to the original IHTM result as baseline. The "user rank" (scale: 1–best, 6–worst) is not predicted by any individual quality metric, and does not align with the measured quality improvements.

6.3 Quantitative Results: Model Quality Assessment

In preparation for our third study, a topic modeling expert ran six different configurations of the modeling process for each input corpus. Three of the configurations used a topic backbone, while the other three did not. Within the two groups, speculative execution was once disabled, and once performed automatically. "Automatically", here, means that the top-rated strategy offered by the system was always accepted. The last two configurations used supervised SpecEx, meaning that the expert inspected the proposed optimizations and selected the most fitting one, performing a semi-supervised optimization.

We chose three different datasets for this study: (1) the presidential debate from the first study, (2) a collection of 125 newsgroup articles from the 20news dataset [43], and (3) 100 news articles from the Associated Press [6]. The insert-order of the documents in both news corpora was determined through randomly shuffling their documents. For each corpus, we let four annotators (two topic modeling experts and two novice users) rank the model outputs created by the expert on a scale from one (best) to six (worst), leading to 72 annotations. The results, presented in Table 3, are averaged over the three corpora, sorted to the annotators' ranking, and show that annotators rate models with a backbone higher than those without. The quality metrics confirm this, reporting higher increases in quality for models with a backbone when compared to an IHTM baseline with no backbone and no SpecEx. Independently of whether a backbone was provided or not, users preferred models with supervised SpecEx over no SpecEx, ranking them 1.0 and 3.0, and 4.17 and 4.42, respectively. They even found that automatic SpecEx deteriorated the model quality and consistently ranked it worse than what the baseline IHTM returned, indicating that optimizing according to numerical values without semantics does not work with current quality measures. Contrary to the users' perception of quality the metrics improve most under automatic SpecEx (91.75% and 14.6%). Nonetheless, the metrics also indicate most improvement of quality through supervised SpecEx when compared to models with no SpecEx, finding a 9.65% improvement if no backbone is specified.

6.4 Discussion and Lessons Learned

Through evaluating our work with three independent measures, we confirmed the competitiveness of IHTM and the effectiveness of our visual analytics workspace. Using these different evaluations, we obtained diverse perspectives on relevant quality criteria. Our quantitative evaluation has established that the quality of topic modeling outputs based on a backbone structure was better received. Furthermore, our evaluation shows that annotators favor optimized topic structures that are computed using a supervised speculative execution over fully-automated optimizations. This, however, stands in contrast to the automatically computed quality measures that indicate that automatic SpecEx yields the better results. More details on the differences in the measured quality over the three configurations of SpecEx can be found in Table 2. The results indicate that no single metric perfectly aligns with the human ranking, validating our claim that users should perform a *multi-objective optimization* instead of focusing on a single metric. Often, metrics track opposing goals: Coherence, for example, is typically high for very small topics and can achieve its best possible score when attributing all documents to their own topic. Topic size, in contrast, aims for large, general topics that provide a good overview. Both metrics can never be optimal at the same time, and it is the user's task to select a middle-ground appropriate for their use-case.

User Rank	Backbone	SpecExec	Measured Rank	Improvement
1.00 (0.00)	**yes**	**supervised**	2.00 (1.00)	67.72% (35.98)
3.00 (1.00)	yes	no	2.75 (0.97)	67.01% (65.84)
3.83 (1.64)	yes	automatic	**1.67** (1.15)	**91.75%** (75.36)
4.17 (1.11)	no	supervised	4.33 (0.58)	9.65% (2.04)
4.42 (1.56)	no	no	6.00 (0.00)	0.00% (0.00)
4.83 (0.94)	no	automatic	4.00 (1.73)	14.6% (11.05)

Table 3: Result quality for six different models, averaged over three corpora. Users ranked the models (scale: 1–best, 6–worst) according to their perception of quality. The measured rank is calculated based on the improvements in quality metric scores. For each value, the standard deviation across the three corpora is shown in parentheses.

In addition, our qualitative study highlights the effectiveness of speculative execution in a visual analytics system, as a mechanism for user-guidance and trust-building. Interviewing users with different backgrounds displayed many different usage patterns of our visual analytics framework. However, regardless of their expertise disciplines, all users approved of our incremental, animated visualization of the model-building process. Almost all users reported that our approach allowed them to understand the process of topic modeling for the first time. They also praised the different possibilities for domain-knowledge incorporation, making use of all three methods (backbone priors, direct topic-tree manipulations, and automatic optimizations) during the model optimization sessions of the studies. Such an explainable machine learning technique does not only support users in adapting the models to their tasks and data, but also opens up a venue for using explorative visual analytics as an educational tool.

7 Conclusion

We have presented a visual analytics framework for mixed-initiative topic model optimization. Based on our novel, explainable topic modeling approach, we visualize every step of the model-building, allowing for the tight integration of the users' feedback and domain-knowledge into the machine learning process. We propose a tailored visual analytics workspace that interactively displays all intermediate results of the topic modeling, allowing users to understand and refine them. In addition to direct manipulations of the built topic-tree, our system enables users to prime the topic modeling algorithm with expected outputs, integrating their own data-semantics into the modeling process. For a targeted optimization, we further introduce speculative execution as a novel concept in visual analytics that acts as a preview mechanism for an efficient user-steerable optimization. We have evaluated our technique based on three independent studies, all confirming the validity and effectiveness of our framework for understanding, diagnosing, and refining topic models. Our work will be made publicly accessible as part of the VisArgue framework: `http://visargue.inf.uni.kn/`.

In our future work, we would like to investigate the potential for transferring the concept of speculative execution in visual analytics to other problem domains. Another goal of our research is to examine other potential model-space visualizations to foster further understanding of machine learning processes, opening up black-box computations. Furthermore, continuing this line of research, we would like to explore other perspectives on the model-building process of topic modeling. Lastly, for better user-guidance, we will be studying which measures best capture the human intuition of topic model quality.

REFERENCES

[1] C. C. Aggarwal and C. Zhai. A survey of text clustering algorithms. In *Mining Text Data*, pp. 77–128. Springer, 2012.

[2] E. Alexander, J. Kohlmann, R. Valenza, M. Witmore, and M. Gleicher. Serendip: Topic model-driven visual exploration of text corpora. In *Proc. IEEE Symp. on Visual Analytics Science and Technology (VAST)*, pp. 173–182, 2014.

[3] D. Andrzejewski, X. Zhu, and M. Craven. Incorporating Domain Knowledge into Topic Modeling via Dirichlet Forest Priors. In *Proc. 26th Int. Conf. on Machine Learning*, pp. 25–32, 2009.

[4] M.-F. Balcan, A. Blum, and S. Vempala. Clustering via similarity functions: Theoretical foundations and algorithms, 2008.

[5] F. Beil, M. Ester, and X. Xu. Frequent term-based text clustering. In *Proc. ACM Conf. on Knowledge Discovery and Data Mining*, pp. 436–442, 2002.

[6] D. M. Blei. Latent Dirichlet Allocation in C. Available at: http://www.cs.columbia.edu/ blei/lda-c/. Last accessed: 31/03/2018.

[7] D. M. Blei. Probabilistic Topic Models. *Communications of the ACM*, 55(4):77–84, 2012. doi: 10.1145/2133806.2133826

[8] D. M. Blei, T. L. Griffiths, and M. I. Jordan. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *J. of the ACM*, 57(2):1–30, 1 2010.

[9] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proc. 23rd Int. Conf. on Machine Learning*, pp. 113–120, 2006. doi: 10.1145/1143844.1143859

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *J. of Machine Learning Research*, 3:993–1022, 2003.

[11] G. Bouma. Normalized (Pointwise) Mutual Information in Collocation Extraction. *Proc. GSCL*, pp. 31–40, 1 2009.

[12] J. Boyd-Graber, Y. Hu, and D. Mimno. Applications of topic models. *Foundations and Trends in Information Retrieval*, 11(2–3):143–296, 2017.

[13] F. Chang and G. A. Gibson. Automatic I/O hint generation through speculative execution. In *Proc. Symp. on Operating Systems Design and Implementation*, pp. 1–14, 1999.

[14] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pp. 288–296, 2009.

[15] D. Cheng, R. Kannan, S. Vempala, and G. Wang. A divide-and-merge methodology for clustering. *ACM Trans. on Database Systems (TODS)*, 31(4):1499–1525, 2006.

[16] J. Chuang, D. Ramage, C. Manning, and J. Heer. Interpretation and trust: Designing model-driven visualizations for text analysis. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems*, pp. 443–452. ACM, 2012.

[17] CNN Political Unit. Transcript: Second presidential debate. Available at: http://politicalticker.blogs.cnn.com/2012/10/16/transcript-second-presidential-debate/. Last accessed: 31/03/2018.

[18] Davies, Mark. Corpus of Historical American English. Available at: http://corpus.byu.edu/coha/. Last accessed: 31/03/2018.

[19] C. Ding, T. Li, and W. Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Computational Statistics & Data Analysis*, 52(8):3913–3927, 2008.

[20] W. Dou, X. Wang, R. Chang, and W. Ribarsky. ParallelTopics: A probabilistic approach to exploring document collections. In *Proc. IEEE Conf. on Visual Analytics Science and Technology*, pp. 231–240, 2011. doi: 10.1109/VAST.2011.6102461

[21] W. Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky. HierarchicalTopics: Visually exploring large text collections using topic hierarchies. *IEEE Trans. on Visualization and Computer Graphics*, 19(12):2002–2011, 2013.

[22] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[23] J. Eisenstein, D. H. Chau, A. Kittur, and E. Xing. TopicViz: Interactive topic exploration in document collections. In *Extended Abstracts of SIGCHI Conf. on Human Factors in Computing Systems*, pp. 2177–2182, 2012.

[24] M. El-Assady, R. Sevastjanova, F. Sperrle, D. Keim, and C. Collins. Progressive Learning of Topic Modeling Parameters: A Visual Analytics Framework. *IEEE Trans. on Visualization and Computer Graphics*, 24(1):382–391, 2018. doi: 10.1109/TVCG.2017.2745080

[25] D. H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2(2):139–172, 1987.

[26] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing. Discourse segmentation of multi-party conversation. In *Proc. Annual Meeting on Assn. for Computational Linguistics*, pp. 562–569, 2003.

[27] A. Ganesan, S. Pan, and J. Chen. LDAExplore : Visualizing Topic Models Generated Using Latent Dirichlet Allocation. *Proc. IUI Workshop on Visual Text Analytics*, 2015.

[28] J. H. Gennari, P. Langley, and D. Fisher. Models of incremental concept formation. *Artificial Intelligence*, 40(1):11–61, 1989.

[29] M. Gleicher. Considerations for Visualizing Comparison. *IEEE Trans. on Visualization and Computer Graphics*, 24(1):413–423, 1 2018. doi: 10.1109/TVCG.2017.2744199

[30] J. Görtler, C. Schulz, D. Weiskopf, and O. Deussen. Bubble treemaps for uncertainty visualization. *IEEE Trans. on Visualization and Computer Graphics*, 24(1):719–728, 2018.

[31] J. A. Guerra-Gomez, M. L. Pack, C. Plaisant, and B. Shneiderman. Visualizing Change over Time Using Dynamic Hierarchies: TreeVersity2 and the StemView. *IEEE Trans. on Visualization and Computer Graphics*, 19(12):2566–2575, 2013.

[32] K. M. Hammouda and M. S. Kamel. Efficient phrase-based document indexing for web document clustering. *IEEE Trans. on Knowledge and Data Engineering*, 16(10):1279–1296, 2004.

[33] E. Hoque and G. Carenini. ConVisIT: Interactive topic modeling for exploring asynchronous online conversations. In *Proc. Int. Conf. on Intelligent User Interfaces*, pp. 169–180, 2015.

[34] Y. Hu, J. Boyd-Graber, B. Satinoff, and A. Smith. Interactive topic modeling. *Machine Learning*, 95(3):423–469, 2014.

[35] Jaegul Choo, Changhyun Lee, C. K. Reddy, and H. Park. UTOPIAN: User-Driven Topic Modeling Based on Interactive Nonnegative Matrix Factorization. *IEEE Trans. on Visualization and Computer Graphics*, (12):1992–2001, 2013. doi: 10.1109/TVCG.2013.212

[36] J. Jagarlamudi, H. Daumé Iii, and R. Udupa. Incorporating lexical priors into topic models. *Proc. Conf. of the European Chapter of the Assn. for Computational Linguistics*, pp. 204–213, 2012.

[37] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann. On close and distant reading in digital humanities: A survey and future challenges. *The Eurographics Association*, pp. 83–103, 2015.

[38] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melancon. Visual analytics: Definition, process, and challenges. In A. Kerren, J. Stasko, J.-D. Fekete, and C. North, eds., *Information Visualization*, vol. 4950 of *Lecture Notes in Computer Science*, pp. 154–175. Springer Berlin Heidelberg, 2008. doi: 10.1007/978--3-540--70956-5_7

[39] M. Kim, K. Kang, D. Park, J. Choo, and N. Elmqvist. TopicLens: Efficient multi-level visual topic exploration of large-scale document collections. *IEEE Trans. on Visualization and Computer Graphics*, 23(1):151–160, Jan. 2017.

[40] D. Kuang, J. Choo, and H. Park. Nonnegative Matrix Factorization for Interactive Topic Modeling and Document Clustering. In *Partitional Clustering Algorithms*, pp. 215–243. 2015.

[41] K. Kummamuru, A. Dhawale, and R. Krishnapuram. Fuzzy co-clustering of documents and keywords. In *Proc. of IEEE Conf. on Fuzzy Systems*, vol. 2, pp. 772–777. IEEE, 2003.

[42] M. J. Kusner, Y. Sun, N. I. Kolkin, and K. Q. Weinberger. From word embeddings to document distances. In *Proc. Int. Conf. on Machine Learning*, pp. 957–966, 2015.

[43] K. Lang. Home Page for 20 Newsgroups Data Set. Available at: http://qwone.com/ jason/20Newsgroups/. Last accessed: 31/03/2018.

[44] J. H. Lau, K. Grieser, D. Newman, and T. Baldwin. Automatic Labelling of Topic Models. *Proc. 49th Annual Meeting of the Assn. for Computational Linguistics*, pp. 1536–1545, 2011.

[45] J. H. Lau, D. Newman, S. Karimi, and T. Baldwin. Best topic word selection for topic labelling. *Proc. Int. Conf. on Computational Linguistics*, pp. 605–613, 2010.

[46] B. Lee, G. G. Robertson, M. Czerwinski, and C. S. Parr. Candidtree: visualizing structural uncertainty in similar hierarchies. *Information Visualization*, 6(3):233–246, 2007.

[47] T. Y. Lee, A. Smith, K. Seppi, N. Elmqvist, J. Boyd-Graber, and L. Findlater. The human touch: How non-expert users perceive, interpret, and fix topic models. *Int. J. of Human-Computer Studies*, 105:28–42, 2017.

[48] S. Liu, X. Wang, M. Liu, and J. Zhu. Towards better analysis of machine learning models: A visual analytics perspective. *Visual Informatics*, 1(1):48–56, 2017.

[49] C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, USA, 1999.

[50] A. K. McCallum and D. Mimno. MALLET: A Machine Learning for Language Toolkit, 2002.

[51] G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, pp. 39–41, 1995.

[52] D. Mimno, W. Li, and A. McCallum. Mixtures of hierarchical topics with Pachinko allocation. In *Proc. of Int. Conf. on Machine Learning*, pp. 633–640. ACM, 2007.

[53] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. *Proc. Conf. on Empirical Methods in Natural Language Processing*, pp. 262–272, 2011.

[54] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proc. Annual Meeting of the Assn. for Computation Linguistics*, pp. 1003–1011. Assn. for Computational Linguistics, 2009.

[55] T. Mühlbacher, H. Piringer, S. Gratzl, M. Sedlmair, and M. Streit. Opening the Black Box: Strategies for Increased User Involvement in Existing Algorithm Implementations. *IEEE Trans. on Visualization and Computer Graphics*, 20(12):1643–1652, 2014.

[56] T. Munzner, F. Guimbretière, S. Tasiran, L. Zhang, Y. Zhou, T. Munzner, F. Guimbretière, S. Tasiran, L. Zhang, and Y. Zhou. TreeJuxtaposer: Scalable Tree Comparison using Focus+ Context with Guaranteed Visibility. *ACM Trans. on Graphics*, 22(3):453, 2003.

[57] S. I. Nikolenko, S. Koltcov, and O. Koltsova. Topic modelling for qualitative studies. *J. of Information Science*, 43(1):88–102, 2017.

[58] H. G. Okuno and A. Gupta. Parallel execution of ops5 in qlisp. In *Proc. Fourth Conf. on Artificial Intelligence Applications*, pp. 268–273. IEEE, 1988.

[59] M. Purver, T. L. Griffiths, K. P. Körding, and J. B. Tenenbaum. Unsupervised topic modelling for multi-party spoken discourse. In *Proc. Annual Meeting of the Assn. for Computational Linguistics*, pp. 17–24, 2006.

[60] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proc. 2009 Conf. on Empirical Methods in Natural Language Processing*, pp. 248–256. ACL, 2006.

[61] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proc. Conf. on Uncertainty in Artificial Intelligence*, pp. 487–494. AUAI Press, 2004.

[62] B. M. Schmidt. Words Alone: Dismantling Topic Models in the Humanities. *J. of Digital Humanities*, 2(1):49–65, 2012.

[63] A. Smith, T. Hawes, and M. Myers. Hiérarchie: Interactive visualization for hierarchical topic models. *Proc. Workshop on Interactive Language Learning, Visualization, and Interfaces*, pp. 71–78, 2014.

[64] H. Stitz, S. Luger, M. Streit, and N. Gehlenborg. Avocado: Visualization of workflow–derived data provenance for reproducible biomedical research. In *Computer Graphics Forum*, vol. 35, pp. 481–490. Wiley Online Library, 2016.

[65] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet Processes. *J. of the American Statistical Association*, 101(476):1566–1581, 2006.

[66] C. Vehlow, T. Reinhardt, and D. Weiskopf. Visualizing Fuzzy Overlapping Communities in Networks. *IEEE Trans. on Visualization and Computer Graphics*, pp. 2486–2495, 2013. doi: 10.1109/TVCG.2013.232

[67] F. Wei, S. Liu, Y. Song, S. Pan, M. X. Zhou, W. Qian, L. Shi, L. Tan, and Q. Zhang. TIARA: A visual exploratory text analytic system. In *Proc. ACM Int. Conf. on Knowledge Discovery and Data Mining*, KDD '10, pp. 153–162. ACM, 2010. doi: 10.1145/1835804.1835827

[68] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. In *Proc. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 267–273, 2003.

[69] Y. Yang, Q. Yao, and H. Qu. VISTopic: A visual analytics system for making sense of large document collections using hierarchical topic modeling. *Visual Informatics*, pp. 40–47, 2017. doi: 10.1016/j.visinf.2017.01.005